Lecture Notes in Artificial Intelligence, Vol. 3533, 2005, pp. 571-578

Predicting Construction Litigation Outcome using Particle Swarm Optimization

Kwokwing Chau

Department of Civil and Structural Engineering, Hong Kong Polytechnic University, Hunghom, Kowloon, Hong Kong
cekwchau@polyu.edu.hk

Abstract. Construction claims are normally affected by a large number of complex and interrelated factors. It is highly desirable for the parties to a dispute to know with some certainty how the case would be resolved if it were taken to court. The use of artificial neural networks can be a cost-effective technique to help to predict the outcome of construction claims, on the basis of characteristics of cases and the corresponding past court decisions. In this paper, a particle swarm optimization model is adopted to train perceptrons. The approach is demonstrated to be feasible and effective by predicting the outcome of construction claims in Hong Kong in the last 10 years. The results show faster and more accurate results than its counterparts of a benching back-propagation neural network and that the PSO-based network are able to give a successful prediction rate of up to 80%. With this, the parties would be more prudent in pursuing litigation and hence the number of disputes could be reduced significantly.

1 Introduction

By its very nature, the construction industry is prone to litigation since claims are normally affected by a large number of complex and interrelated factors. The disagreement between the involving parties can arise from interpretation of the contract, unforeseen site conditions, variation orders by the client, acceleration and suspension of works, and so on. The main forums for the resolution of construction disputes are mediation, arbitration, and the courts. However, the consequence of any disagreements between the client and the contractor may be far reaching. It may lead to damage to the reputation of both sides, as well as inefficient use of resources and higher costs for both parties through settlement. The litigation process is usually very expensive since it involves specialized and complex issues. Thus, it is the interest of all the involving parties to minimize or even avoid the likelihood of litigation through conscientious management procedure and concerted effort.

It is highly desirable for the parties to a dispute to know with some certainty how the case would be resolved if it were taken to court. This would effectively help to significantly reduce the number of disputes that would need to be settled by the much more expensive litigation process. The use of artificial neural networks can be a cost-effective technique to help to predict the outcome of construction claims, on the basis of characteristics of cases and the corresponding past court decisions. It can be used to identify the hidden relationships among various interrelated factors and to mimic decisions that were made by the court.

During the past decade, the artificial neural networks (ANN), and in particular, the feed forward backward propagation perceptrons, are widely applied in different fields [1-2]. It is claimed that the multi-layer perceptrons can be trained to approximate and accurately generalize virtually any smooth, measurable function whilst taking no prior assumptions concerning the data distribution. Characteristics, including built-in dynamism in forecasting, data-error tolerance, and lack of requirements of any exogenous input, render it attractive for use in various types of prediction. Although the back propagation (BP) algorithm is commonly used in recent years to perform the training task, some drawbacks are often encountered in the use of this gradient-based method. They include: the training convergence speed is very slow; it is easily to get stuck in a local minimum. Different algorithms have been proposed in order to resolve these drawbacks, yet the results are still not fully satisfactory [3-5].

Particle swarm optimization (PSO) is a method for optimizing hard numerical functions based on metaphor of human social interaction [6-7]. Although it is initially developed as a tool for modeling social behavior, the PSO algorithm has been recognized as a computational intelligence technique intimately related to evolutionary algorithms and applied in different areas [8-11].

In this paper, a PSO-based neural network approach for prediction of the outcome of construction litigation in Hong Kong is developed by adopting PSO to train multi-layer perceptrons, on the basis of characteristics of real cases and court decisions in the last 10 years.

2 Nature of Construction Disputes

The nature of construction activities is varying and dynamic, which can be evidenced by the fact that no two sites are exactly the same. Thus the preparation of the construction contract can be recognized as the formulation of risk allocation amongst the involving parties: the client, the contractor, and the engineer. The risks involved include the time of completion, the final cost, the quality of the works, inflation, inclement weather, shortage of materials, shortage of plants, labor problems, unforeseen ground conditions, site instructions, variation orders, client-initiated changes, engineer-initiated changes, errors and omissions in drawings, mistakes in specifications, defects in works, accidents, supplier delivery failure, delay of schedule by subcontractor, poor workmanship, delayed payment, changes in regulations, third-party interference, professional negligence, and so on.

Prior to the actual construction process, the involving parties will attempt to sort out the conditions for claims and disputes through the contract documents. However, since a project usually involves thousands of separate pieces of work items to be integrated together to constitute a complete functioning structure, the potential for honest misunderstanding is extremely high. The legislation now in force requires that any disputes incurred have to be resolve successively by mediation, arbitration, and the courts [12].

3 Multi-layer Feed-forward Perceptron

A multi-layer feed-forward perceptron represents a nonlinear mapping between input vector and output vector through a system of simple interconnected neurons. It is fully connected to every node in the next and previous layer. The output of a neuron is scaled by the connecting weight and fed forward to become an input through a nonlinear activation function to the neurons in the next layer of network. In the course of training, the perceptron is repeatedly presented with the training data. The weights in the network are then adjusted until the errors between the target and the predicted outputs are small enough, or a pre-determined number of epochs is passed. The perceptron is then validated by presenting with an input vector not belonging to the training pairs. The training processes of ANN are usually complex and high dimensional problems. The commonly used gradient-based BP algorithm is a local search method, which easily falls into local optimum point during training.

4 Particle Swarm Optimization (PSO)

Particle swarm optimization (PSO) is an optimization paradigm that mimics the ability of human societies to process knowledge. It has roots in two main component methodologies: artificial life (such as bird flocking, fish schooling and swarming); and, evolutionary computation. The key concept of PSO is that potential solutions are flown through hyperspace and are accelerated towards better or more optimum solutions.

4.1 PSO Algorithm

PSO is a populated search method for optimization of continuous nonlinear functions resembling the movement of organisms in a bird flock or fish school. Its paradigm can be implemented in a few lines of computer code and is computationally inexpensive in terms of both memory requirements and speed. It lies somewhere between evolutionary programming and genetic algorithms. As in evolutionary computation paradigms, the concept of fitness is employed and candidate solutions to the problem are termed particles or sometimes individuals. A similarity between PSO and a genetic algorithm is the initialization of the system with a population of random solutions. Instead of employing genetic operators, the evolution of generations of a population of these individuals in such a system is by cooperation and competition among the individuals themselves. Moreover, a randomized velocity is assigned to each potential solution or particle so that it is flown through hyperspace. The adjustment by the particle swarm optimizer is ideally similar to the crossover operation in genetic algorithms whilst the stochastic processes are close to evolutionary programming. The stochastic factors allow thorough search of spaces between regions that are spotted to be relatively good whilst the momentum effect of modifications of the existing velocities leads to exploration of potential regions of the problem domain.

There are five basic principles of swarm intelligence: (1) proximity; (2) quality; (3) diverse response; (4) stability; and, (5) adaptability. The n-dimensional space calculations of the PSO concept are performed over a series of time steps. The population is responding to the quality factors of the previous best individual values and the previous best group values. The allocation of responses between the individual and group values ensures a diversity of response. The principle of stability is adhered to since the population changes its state if and only if the best group value changes. It is adaptive corresponding to the change of the best group value.

In essence, each particle adjusts its flying based on the flying experiences of both itself and its companions. It keeps track of its coordinates in hyperspace which are associated with its previous best fitness solution, and also of its counterpart corresponding to the overall best value acquired thus far by any other particle in the population. Vectors are taken as presentation of particles since most optimization problems are convenient for such variable presentations. The stochastic PSO algorithm has been found to be able to find the global optimum with a large probability and high convergence rate. Hence, it is adopted to train the multi-layer perceptrons, within which matrices learning problems are dealt with.

4.2 Adaptation to Network Training

A three-layered preceptron is chosen for this application case. Here, $W^{[1]}$ and $W^{[2]}$ represent the connection weight matrix between the input layer and the hidden layer, and that between the hidden layer and the output layer, respectively. When a PSO is employed to train the multi-layer preceptrons, the i-th particle is denoted by

$$W_i = \{W_i^{[1]}, W_i^{[2]}\} \tag{1}$$

The position representing the previous best fitness value of any particle is recorded and denoted by

$$P_i = \{P_i^{[1]}, P_i^{[2]}\} \tag{2}$$

If, among all the particles in the population, the index of the best particle is represented by the symbol b, then the best matrix is denoted by

$$P_b = \{P_b^{[1]}, P_b^{[2]}\} \tag{3}$$

The velocity of particle i is denoted by

$$V_i = \{V_i^{[1]}, V_i^{[2]}\} \tag{4}$$

If m and n represent the index of matrix row and column, respectively, the manipulation of the particles are as follows

$$\begin{aligned} V_i^{[j]}(m,n) = V_i^{[j]}(m,n) + r\alpha[P_i^{[j]}(m,n) - W_i^{[j]}(m,n)] \\ + s\beta[P_b^{[j]}(m,n) - W_i^{[j]}(m,n)] \end{aligned} \tag{5}$$

and

$$W_i^{[j]} = W_i^{[j]} + V_i^{[j]} \tag{6}$$

where j = 1, 2; m = 1, …, M_j; n= 1, …, N_j; M_j and N_j are the row and column sizes of the matrices W, P, and V; r and s are positive constants; α and β are random numbers in the range from 0 to 1. Equation (5) is employed to compute the new velocity of the particle based on its previous velocity and the distances of its current position from the best experiences both in its own and as a group. In the context of social behavior, the cognition part $r\alpha[P_i^{[j]}(m,n) - W_i^{[j]}(m,n)]$ represents the private thinking of the particle itself whilst the social part $s\beta[P_b^{[j]}(m,n) - W_i^{[j]}(m,n)]$ denotes the collaboration among the particles as a group. Equation (6) then determines the new position according to the new velocity [6-7].

The fitness of the i-th particle is expressed in term of an output mean squared error of the neural networks as follows

$$f(W_i) = \frac{1}{S}\sum_{k=1}^{S}\left[\sum_{l=1}^{O}\{t_{kl} - p_{kl}(W_i)\}^2\right] \tag{7}$$

where f is the fitness value, t_{kl} is the target output; p_{kl} is the predicted output based on W_i; S is the number of training set samples; and, O is the number of output neurons.

5 The Study

The system is applied to study and predict the outcome of construction claims in Hong Kong. The data from 1991 to 2000 are organized case by case and the dispute characteristics and court decisions are correlated. Through a sensitivity analysis, 13 case elements that seem relevant in courts' decisions are identified. They are, namely, type of contract, contract value, parties involved, type of plaintiff, type of defendant, resolution technique involved, legal interpretation of contract documents, misrepresentation of site, radical changes in scope, directed changes, constructive changes, liquidated damages involved, and late payment.

Some of the 13 case elements can be expressed in binary format; for example, the input element 'liquidated damages involved' receives a 1 if the claim involves liquidated damages or a 0 if it does not. However, some elements are defined by several alternatives; for example, 'type of contract' could be remeasurement contract, lump sum contract, or design and build contract. These elements with alternative answers are split into separate input elements, one for each alternative. Each alternative is represented in a binary format, such as 1 for remeasurement contract and 0 for the others if the type of contract is not remeasurement. In that case, only one of these input elements will have a 1 value and all the others will have a 0 value. In this way, the 13 elements are converted into an input layer of 30 neurons, all expressed in binary format. Table 1 shows examples of the input neurons for cases with different types of contract. The court decisions are also organized in an output layer of 6 neurons expressed in binary format corresponding to the 6 elements: client, contractor, engineer, sub-contractor, supplier, and other third parties.

In total, 1105 sets of construction-related cases were available, of which 550 from years 1991 to 1995 were used for training, 275 from years 1996 to 1997 were used for testing, and 280 from years 1998 to 2000 were used to validate the network results with the observations. It is ensured that the data series chosen for training and validation comprised balanced distribution of cases.

Table 1. Examples of the input neurons for cases with different types of contract

Input neuron	Cases		
	Remeasurement	Lump sum	Design and build
Type of contract - remeasurement	1	0	0
Type of contract - lump sum	0	1	0
Type of contract – design and build	0	0	1

Sensitivity analysis is performed to determine the best architecture, with variations in the number of hidden layers and number of hidden neurons. The final perceptron has an input layer with thirty neurons, a hidden layer with fifteen neurons, and output layer with six neurons. In the PSO-based perceptron, the number of population is set to be 40 whilst the maximum and minimum velocity values are 0.25 and -0.25 respectively.

6 Results and Discussions

The PSO-based multi-layer ANN is evaluated along with a commonly used standard BP-based network. In order to furnish a comparable initial state, the training process of the BP-based perceptron commences from the best initial population of the corresponding PSO-based perceptron. Figure 1 shows the relationships between the normalized mean square error and fitness evaluation time during training for PSO-based and BP-based perceptrons. Table 2 shows comparisons of the results of network for the two different perceptrons.

The fitness evaluation time here for the PSO-based perceptron is equal to the product of the population with the number of generations. It is noted that testing cases of the PSO-based network are able to give a successful prediction rate of up to 80%, which is much higher than by pure chance. Moreover, the PSO-based perceptron exhibits much better and faster convergence performance in the training process as well as better prediction ability in the validation process than those by the BP-based perceptron. It can be concluded that the PSO-based perceptron performs better than the BP-based perceptron. It is believed that, if the involving parties to a construction dispute become aware with some certainty how the case would be resolved if it were taken to court, the number of disputes could be reduced significantly.

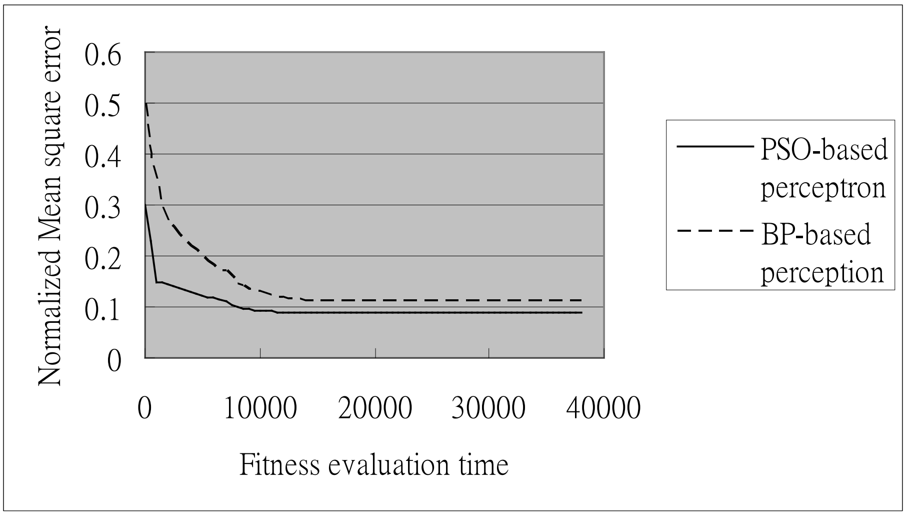

Fig. 1. Relationships between the normalized mean square error and fitness evaluation time during training for PSO-based and BP-based perceptrons

Table 2. Comparison of prediction results for outcome of construction litigation

Algorithm	Training		Validation	
	Coefficient of correlation	Prediction rate	Coefficient of correlation	Prediction rate
BP-based	0.956	0.69	0.953	0.67
PSO-based	0.987	0.81	0.984	0.80

7 Conclusions

This paper presents a PSO-based perceptron approach for prediction of outcomes of construction litigation on the basis of the characteristics of the individual dispute and the corresponding past court decisions. It is demonstrated that the novel optimization algorithm, which is able to provide model-free estimates in deducing the output from the input, is an appropriate prediction tool. The final network presented in this study is recommended as an approximate prediction tool for the parties in dispute, since the rate of prediction is up to 80%, which is much higher than chance. It is, of course, recognized that there are limitations in the assumptions used in this study. Other factors that may have certain bearing such as cultural, psychological, social, environmental, and political factors have not been considered here. Nevertheless, it is shown from the training and verification simulation that the prediction results of outcomes of construction litigation are more accurate and are obtained in relatively short computational time, when compared with the commonly used BP-based perceptron. Both the above two factors are important in construction management. It can be concluded that the PSO-based perceptron performs better than the BP-based perceptron.

8 Acknowledgement

This research was supported by the Central Research Grant of Hong Kong Polytechnic University (G-T592) and the Internal Competitive Research Grant of Hong Kong Polytechnic University (A-PE26).

References

1. Arditi, D., Oksay, F.E., Tokdemir, O.B.: Predicting the Outcome of Construction Litigation Using Neural Networks. Computer-Aided Civil and Infrastructure Engineering **13(2)** (1998) 75-81
2. Thirumalaiah, K., Deo, M.C.: River Stage Forecasting Using Artificial Neural Networks. Journal of Hydrologic Engineering, ASCE **3(1)** (1998) 26-32
3. Govindaraju, R., Rao, A. (Ed.): Artificial Neural Networks in Hydrology. Kluwer Academic Publishers, Dordrecht (2000)
4. Liong, S.Y., Lim, W.H., Paudyal, G.N.: River Stage Forecasting in Bangladesh: Neural Network Approach. Journal of Computing in Civil Engineering, ASCE **14(1)** (2000) 1-8
5. Chau, K.W., Cheng, C.T.: Real-time Prediction of Water Stage with Artificial Neural Network Approach. Lecture Notes in Artificial Intelligence, **2557** (2002) 715-715
6. Kennedy, J., Eberhart, R.: Particle Swarm Optimization. Proceedings of the 1995 IEEE International Conference on Neural Networks. Perth (1995) 1942-1948
7. Kennedy, J.: The Particle Swarm: Social Adaptation of Knowledge. Proceedings of the 1997 International Conference on Evolutionary Computation. Indianapolis (1997) 303-308
8. Clerc, M., Kennedy, J.: The Particle Swarm—Explosion, Stability, and Convergence in a Multidimensional Complex Space. IEEE Transactions on Evolutionary Computation **6(1)** (2002) 58-73
9. Kennedy, J., Eberhart, R., Shi, Y.: Swarm Intelligence. Morgan Kaufmann Publishers, San Francisco (2001)
10. Chau, K.W.: River Stage Forecasting with Particle Swarm Optimization. Lecture Notes in Computer Science **3029** (2004) 1166-1173
11. Chau, K.W.: Rainfall-Runoff Correlation with Particle Swarm Optimization Algorithm. Lecture Notes in Computer Science **3174** (2004) 970-975
12. Chau, K.W.: Resolving Construction Disputes by Mediation: Hong Kong Experience. Journal of Management in Engineering, ASCE **8(4)** (1992) 384-393